CMS Requirements for Batch Queuing Systems
May 20, 2004

Introduction:
US-CMS will spend the time between now and 2007 building up the capacity and functionality of the Tier-1 computing center at FNAL in preparation for LHC data. One of the core services offered by a Tier-1 center is batch processing. The CMS requirements up to now could have been easily satisfied by a number of batch queuing systems, due to the limited scale and complexity of CMS computing. As we approach a significant ramp in capacity and service, we need to assess our batch processing requirements.

The US-CMS batch processing needs are divided into three categories: functionality, scalability, and support. Within those categories the needs are divided between required features and desired features. The items listed under desired functionality would simplify the implementation of facility services and improve the overall quality of service, but the presence or absence of these features would not disqualify a potential batch processing solution.

Functionality:
The batch queuing system adopted by US-CMS must have the ability to richly describe policies used to set user priorities. US-CMS is expecting batch requests from the local analysis community, Open Science Grid Users, and LHC Computing Grid users. We need to be able to specify the processing priority of a user based on VO, role, and application. We would like to be able to define the global priority of a virtual organization and then subdivide the priority based on role and activity. As an example, we would like to ensure that 65% of all resources are available for use by the CMS VO members. We would then like to provide one quarter of those resources to members doing analysis on the UAF, one quarter to incoming grid analysis users, and one half to CMS controlled event reconstruction. It would be desirable to allow VO administrators to change their internal policies without system administrator intervention.

The batch queuing system must have the ability to enforce quotas on resource utilization: CPU usage, wall clock time usage, and memory usage. US-CMS is expecting many users from outside the site and needs to be able to enforce site policy.

The batch queuing system must be able to a match request requirements and system requirements. US-CMS systems are on a three-year replacement cycle; worker nodes will vary from new to nearing the end of their life cycle. The batch queue should be able to match an application to worker node based on memory, CPU speed, OS version, or administrator defined quantity. Some CMS systems may have better access to data, or other facility services, than others and administrators and users may wish to preferentially select systems based on definable criteria.

Jobs must be able to be submitted and scheduled through multiple grid interfaces to the same cluster of physical resources. US-CMS will support users from the LHC

Computing Grid (LCG), the Open Science Grid (OSG), and local FNAL users. Ideally a common agreed upon interface to processing resources would be deployed. It is likely we will have to maintain multiple interfaces for some time.

The following are desired pieces of functionality, but the lack of them should not disqualify a solution.

1. Job pre-emption is a desirable feature. Past experience indicates that check pointing the CMS application is difficult and has not maintained the needed focus of the software development team. However, pre-empting an application and forcing it into an idle state, while a shorter higher priority application runs could simplify the interactions between production and analysis jobs.
2. Failover protection of critical components is a desirable feature. There are typically several processes running in order to maintain the functionality of the queuing system. Since processing services are critical components for CMS and will eventually require 24/7 support, automatic failover protection that could reduce the operations load is attractive.

Scalability:
US-CMS expects to support 500-700 worker nodes running 2-4 processes each at the start of the experiment in 2007. At the start of high-luminosity running, scheduled for 2009, the expectation is for twice that many. The batch queuing system should be able to schedule and manage 2000 running processes in 2007 and 4000 in 2009.

It is possible to install multiple instances of the batch infrastructure and partition the farm, if the scaling requirements cannot be met. In terms of optimal user of resources, it is preferable if the scheduling services can be made to scale.

Support:
In the case of the batch system product, there are two identified types of support: product development support and the local effort required to support the product. US-CMS needs to be aware of the effort required in each.

CMS is a long-term project, and while our architecture should allow the replacement of components, we expect to maintain facility components for several years after adoption. The batch system chosen should have sufficient central support and development effort to expect a viable and maintained production until at least 2008. We acknowledge that "sufficient" support is a subjective term and a true guarantee of 4 years support is probably impossible in practice.

The other element of support is the effort required to interface the product to grid services. The batch queuing system used in CMS will be expected to interface to the current generation of grid services and prototype grid interfaces currently in development. In the past US-CMS has had to expend effort to develop and maintain the interfaces to the queuing system used, because it was unique. This effort was not required at the US Tier-2 centers, which opted to use more common queuing systems for

which the interfacing effort had already been done. US-CMS will not disqualify solutions for not having broad adoption, but the effort required to locally support the product will be considered in the evaluation process.